

<p>BEACON HR/PAYROLL IMPLEMENTATION PROJECT INTERFACE STRATEGY</p>
--

TABLE OF CONTENTS

1. OVERVIEW	3
2. DESCRIPTION.....	3
3. SAP INTERFACING TOOLS AND TECHNIQUES:	3
3.1. IN DETAIL	4
3.2. Batch Input (BI) & Direct Input (DI) programs:	5
3.3. Remote Function Call (RFC):	6
3.4. Electronic Data Interchange (EDI):.....	7
3.5. Application Link Enabling (ALE):	7
3.6. Business API (BAPI):	8
3.7. SAP Open integration and application platform (SAP Exchange Infrastructure – SAP NetWeaver):.....	9
3.8. SAP Connectors:.....	10
3.9. Internet Transaction Server (ITS):.....	10
3.10. CPI-C communication:	11
4. WORK APPROACH:	12
4.1. Prepare Detail Plan for Interface development	12
4.2. Identify the Integration Points (Scope)	12
4.3. Complete Detail Requirement Analysis	12
4.4. Determine the Appropriate Technique	13
4.5. Low-level communication techniques.....	13
4.6. Develop interface programs/objects and complete unit test.....	13
4.7. Complete Integration test	14
4.8. Data Validation	14
4.9. Perform Reconciliation Process	14
5. SAMPLE ANALYSIS DATA GATHERED:.....	14

1. OVERVIEW

The purpose of this document is to describe the BEACON project's detailed interface development strategy.

This document takes a closer look at these interfacing techniques, effective ways and scenarios to employ them, and their pros & cons. At the end of the document you can also find links to various resources to get more details about each of these interfacing technologies.

2. DESCRIPTION

One of the primary challenges of SAP implementation process is integrating SAP with other applications. These interfaces can be inbound or outbound from SAP and enable the State to link together all related systems.

SAP provides the basic infrastructure needed to bridge these systems by providing various tools, techniques & technologies for the integration.

3. SAP INTERFACING TOOLS AND TECHNIQUES:

SAP continually develops and updates its tools and techniques to support emerging industry standard interfacing technologies from time to time.

The following are the techniques currently supported by SAP for both batch and real-time interfacing:

- Batch Data Communication (BDC)
- Batch Input (BI) & Direct Input (DI) programs
- Remote Function Call (RFC)
- Electronic Data Interchange (EDI)
- Application Link Enabling (ALE)
- Business Application Programming Interface (BAPI)
- SAP Open integration and application platform (SAP NetWeaver)
- SAP Connectors
- Internet Transaction Server (ITS)
- Exchange Infrastructure (XI).

Low-level communication techniques:

- CPI-C Communication
- Platform specific development libraries for SNA & TCP/IP communication

In addition to the above technologies, a number of third party interfacing tools are available to integrate SAP with Non-SAP systems.

3.1. IN DETAIL

3.1.1. Batch Data Communication (BDC):

Batch Data Communication (BDC) is the oldest batch interfacing technique that SAP provided since the early versions of R/3. BDC is not a typical integration tool, in the sense that, it can be only be used for uploading data into R/3 and so it is not bi-directional.

BDC works on the principle of simulating user input for transactional screen, via an ABAP program. Typically the input comes in the form of a flat file. The ABAP program reads this file and formats the input data screen by screen into an internal table (BDCDATA). The transaction is then started using this internal table as the input and executed in the background.

Since the BDC technique invokes the standard SAP transaction, which the normal user would invoke interactively, all the input field validations, translations and flow are intact and behaves in accordance with the expected process flow (Please note that this is does not hold good in a few cases, in which the Dialogue module checks whether the Transaction is being executed interactively or in the BDC Batch mode in order to control the screen flow, e.g. getting the confirmation from the end-user).

This flip side of it is performance hit. Since BDC has to go through all the Transactional screens and validations, most of the time is spent by the system in loading the DYNPRO (Dialog screens) as opposed to doing the actual business function.

There are two ways to run BDC programs:

- Using CALL TRANSACTION
- Using Batch Input Sessions.

In 'Call Transaction', the transactions are triggered at the time of processing itself and so the ABAP program must do the error handling. It can also be used for real-time interfaces and custom error handling & logging features.

Whereas in Batch Input Sessions, the ABAP program creates a session with all the transactional data, and this session can be viewed, scheduled and processed (using Transaction SM35) at a later time. The latter technique has a built-in error processing mechanism too.

The BDC Transaction recorder (Transaction SHDB) helps in recording the user inputs and the screen flow of a transaction and can generate ABAP code to create BDC sessions. With this facility, the programmer no longer needs to dig through the attributes to note down the field names, program names and screen numbers to create a BDC program.

Though BDC is simple enough to develop, it is not very efficient when compared to other interfacing options like Batch Input (BI), Direct Input (DI), RFC, and BAPI etc...

SAP has been introducing more and more BI / DI programs & BAPIs in the new releases, in order to provide better integration at the business object level and at a higher speed.

3.2. Batch Input (BI) & Direct Input (DI) programs:

These are SAP supplied standard ABAP programs for loading master and transaction data into SAP. Both these types of programs take a flat file as the input and load the data into the R/3 system.

Batch Input (BI) programs still use the classical BDC approach but don't require an ABAP program to be written to format the BDCDATA. The user has to format the data using predefined structures and store it in a flat file. The BI program then reads this and invokes the transaction mentioned in the header record of the file.

Direct Input (DI) programs work exactly similar to BI programs. But the only difference is, instead of processing screens they validate fields and directly load the data into tables using standard function modules. For this reason, DI programs are much faster (RMDATIND - Material Master DI program works at least 5 times faster²) than the BDC counterpart and so ideally suited for loading large volume data.

DI programs are not available for all application areas. Only a limited number of DI programs have been supplied by SAP.

SAP supplied Direct Input Programs (Sample List):

Program	Description
RMDATIND	MM: Create Material master data
RFBIBL00	FI: Create Account documents
RVAFSS00	SD: Invoicing external transactions
RAALTD11	AM: Create Asset Account documents
RKEVEXT0	CO-PA: External data transfer to CO

The Direct Input sessions can be created, viewed, scheduled, processed and restarted using Transaction BMV0 or using Program RBMVSHOW.

3.3. Remote Function Call (RFC):

Remote Function Call (RFC) in SAP R/3 is based on UNIX's Remote Procedure Call (RPC). RFC enables the real-time link between different R/3 instances and can connect SAP to Non-SAP systems.

With Remote Function Call, both inbound (RFCServer) and outbound (RFCClient) interfaces are possible. SAP also provides a toolkit called RFC API for non-SAP systems to interact with SAP's Function modules remotely.

SAP maintains a central repository of interoperable system definitions called "Destinations" (Transaction SM59).

The following types of connections are possible with RFC Destinations.

- Connection to R/3 system
- Connection to R/2 system
- External program connection via TCP/IP
- External program connection via SNA or APPC
- RFC via special ABAP/4 drivers.

From release 3.0 onwards, SAP has introduced a new version of RFC called Transactional RFC (tRFC). tRFC makes sure that the function remote calls are transactional in nature that is, safe, independent of the availability of the RFC server and being called only once. SAP provides support of both Synchronous and Asynchronous RFC calls.

RFC is a very powerful tool for interfacing with SAP system in real time. Most other interfacing techniques like ALE, EDI, and BAPI etc... are inherently based on the RFC technology.

3.4. Electronic Data Interchange (EDI):

Electronic Data Interchange (EDI) is the industry standard format for exchanging Business data. SAP uses a neutral format called Intermediate Document (IDoc) for communicating EDI & ALE messages.

The IDoc interface contains data structures and processing logic for each of the business transaction. These are grouped as Message Types. The external EDI Server is responsible for receiving the EDI messages sent by the trading partners. These EDI messages are then mapped onto the IDoc structure fields and stored in the IDoc interface layer by SAP.

EDI/IDoc supports both inbound and outbound interfaces. The IDocs received through the EDI Server can be configured to automatically get processed, or can be stored in the IDoc interface layer for a future manual or scheduled processing.

The IDoc interface layer also has a powerful error handling, reporting, monitoring and reprocessing mechanism built into it (Transaction WEDI). IDoc is also tightly integrated with the Workflow system of SAP for initiating other business tasks upon receiving/sending EDI messages.

The IDoc interface can be employed for non-SAP applications to load or extract data from SAP system. But typically, EDI / IDoc is engaged when SAP system is to be integrated with any of its business partners (Customers, Suppliers and other partners) for carrying out standard business transactions (like Create Sales orders, Sending Purchase orders to Suppliers, etc...).

3.5. Application Link Enabling (ALE):

Application Link Enabling (ALE) is a middleware tool, which is part of SAP's Business framework Architecture (BFA). ALE can effectively be used for loosely coupled Asynchronous communication between SAP systems and with non-SAP systems. ALE is based on the distributed processing architecture and hence integrates applications with the local database rather than centralized one.

ALE messages are based on the Intermediate Document (IDoc) format, similar to the EDI counterpart. Though ALE can be used to interface with non-SAP systems, usually it is used to distribute Master & Transactional data between SAP R/3 instances.

The ALE distribution can be based on,

- Business APIs
- IDoc Message Types.

Distributions based on IDoc Message types (example, MATMAS / MATCOR for Material Master data & ORDERS / ORDCHG for Customer master etc...) are typically used in SAP 3.X releases.

The order of the day (since release 4.0A) and future is ALE distribution through BAPIs. The main advantage of ALE is that it can work between different releases of R/3 systems. So, even after a release (version) upgrade, ALE links continue to work as is. Since ALE is based on the IDoc processing model, error handling becomes easy.

3.6. Business API (BAPI):

Business API (BAPI) is the most important component of SAP Business Framework. BAPI enables SAP and third party applications to interact and integrate with each other at the Business Object / Process level.

Till R/2, SAP was deep into its not-so-friendly interface and architecture, which it inherited from the IBM Mainframe based parent system. But now it has come up with truly platform independent business object interfaces using BAPIs. Business Objects and BAPIs provide an object-oriented view of SAP's business functionality (Transaction BAPI).

BAPIs are implemented using remotely callable Function modules. These RFCs are assigned as a callable method of a Business Object in the Business Object Repository (BOR). All BAPIs are independent of the screen and hence can work in the background without user intervention. BAPI supports both inbound and outbound processing. Since it follows object-oriented approach, SAP has positioned itself to support interfacing through BAPIs in the future.

3.7. SAP Open integration and application platform (SAP Exchange Infrastructure – SAP NetWeaver):

SAP Exchange Infrastructure (SAP XI) is SAP's platform for process integration based on the exchange of XML messages. SAP Exchange Infrastructure (SAP XI) is the component within SAP NetWeaver, empowering cross-component process management.

SAP Exchange Infrastructure

- Provides a technical infrastructure for XML-based message exchange in order to connect SAP components with each other, as well as with non-SAP components
- Delivers business-process and integration knowledge to the customer, in the form of SAP's predefined business scenarios
- Provides an integrated toolset for building new business scenarios by defining and maintaining all integration-relevant information ("shared collaboration knowledge").

The most important new capabilities of SAP XI 3.0 are:

3.7.1. Cross-Component Business Process Management

Cross-component business processes can be modeled and stored in the Integration Repository. A Business Process Engine running tightly integrated with the rest of the SAP XI runtime environment executes business processes.

3.7.2. B2B Integration

Collaboration partner profile data can be maintained in the Integration Directory, and communication with those business partners is based on this data. A Partner Connectivity Kit allows the integration of business partners not using SAP XI or other integration solution.

3.7.3. Industry standard support

RosettaNet communication with business partners is supported through a RosettaNet adapter. The RosettaNet adapter is part of an overall package to enable SAP solutions to participate in RosettaNet integration. The package includes necessary mappings.

3.7.4. Adapter Framework and certification of adapters

A JCA-based adapter framework allows the integration of adapters provided by SAP as well as by partners. The integration of partner adapters will be certifiable.

3.8. SAP Connectors:

There are a number of connectors supplied by SAP for integrating R/3 with different applications and technologies.

- SAP Business Connectors
- SAP Java Connectors
- SAP .NET Connectors
- SAP DCOM Connectors.

The Java, .NET and DCOM connectors are tool kits, which are functionally similar, to enable interoperability between SAP and other software. These toolkits typically come with middleware APIs to support RFC (synchronous, transactional & queued), BAPIs, and IDocs. These connectors are ideal for custom interface development for the in-house integration needs.

SAP Business Connectors is a step above the technology connectors. Business Connectors allow integration of local and application over the net, with open standards (like XML etc...). The BAPI and IDoc interfaces are made available to partners over the net using secured XML based services. In addition, Business Connectors enable to expose additional Web Services over the Internet for the business partners to get information deeply buried into your ERP system.

3.9. Internet Transaction Server (ITS):

Internet Transaction Server (ITS) is the transaction processing system that connects the web HTTP server and the standard R/3 system. ITS is available since SAP release 3.1G. All the Internet Application Components (IAC) of SAP will be transacted through ITS.

Using ITS & IAC the standard R/3 transactions can be performed through the World Wide Web.

Some of the business components that are pre-web-enabled by SAP are,

- HR Employee Self-Service (ESS)
- Components in SAP Retail Store
- SAP B2B Procurement.

It is also possible to web-enable the custom transactions and other standard transactions using ITS, using the following tools and components supplied by SAP.

- SAP@Web Studio

- SAP GUI for HTML
- Web RFC
- Web Reporting
- Web Transactions.

These components support generation of dynamic HTML, HTMLBusiness, JavaScript, Common Gateway Interface (CGI), Microsoft Information Server API (ISAPI), Netscape Server API (NSAPI) and can interact with Java, C++ or VB programs.

3.10. CPI-C communication:

Common Programming Interface – Communication (CPI-C) is a direct program-to-program communication protocol. Those who want to develop simple but proprietary protocol can use this technique. Remote Function Call (RFC) is a simplified version of CPI-C, so that the programmer need not worry about the communication protocols and rather can concentrate on communicating business information between programs.

CPI-C allows an ABAP program to set up a connection to one or more external application programs in order to send data directly to these programs.

Few CPI-C ABAP Statements:

CPI-C Call in ABAP	Explanation
COMMUNICATION INIT	Initialize connection
COMMUNICATION ALLOCATE	Set up connection
COMMUNICATION ACCEPT	Accept connection
COMMUNICATION SEND	Send Data
COMMUNICATION RECEIVE	Receive data
COMMUNICATION DEALLOCATE	Close the connection

In short, CPI-C is the low-level stuff that hard-core protocol developers will be interested in. It is not meant for Business-Object-level or Application-level integration.

4. WORK APPROACH:

The interface development or integrating SAP with the other applications process require the completion of seven primary activities:

1. Prepare detail plan for Interface development
2. Identify the Integration points (scope)
3. Complete detail requirement analysis
4. Determine the appropriate technique
5. Develop interface programs/objects and complete unit test
6. Complete integration test which includes background job scheduling, Back-up/recovery of data files and FTP scenarios
7. Data validation on both applications.

4.1. Prepare Detail Plan for Interface development

Using a planning tool such as Microsoft Project or Excel, the development team will prepare a detailed interface development object list and plan. The project leadership team will approve the plan, which will contain detail steps along with dates and resources.

4.2. Identify the Integration Points (Scope)

In SAP, a business object is a general category for data that defines items such as a personal data master, a material data master, payroll balances, general ledger account master, cost center or department master, fund master, benefit elections, or organizational units. The first step will be to identify which business objects/Applications are required for data exchange between SAP and other systems.

4.3. Complete Detail Requirement Analysis

Gathering detail requirements such as Data file format, Frequency of execution, dependencies, Back-up/recovery, FTP requirements are an important step in the planning and design of the interface development strategy. We will lay down criteria for determining/approving the interface development requirements during workshops. Detail requirement analysis will be performed separately for each integration point. Detail requirements analysis allows the consultants to establish benchmarks and appropriate processing times.

4.4. Determine the Appropriate Technique

For each interface development, the development team will work with the functional team members to select one of the following the techniques (both batch and real-time interfaces):

- Batch Data Communication (BDC)
- Batch Input (BI) & Direct Input (DI) programs
- Remote Function Call (RFC)
- Electronic Data Interchange (EDI)
- Application Link Enabling (ALE)
- Business Application Programming Interface (BAPI)
- SAP Open integration and application platform (SAP NetWeaver)
- SAP Connectors
- Internet Transaction Server (ITS).

4.5. Low-level communication techniques

- CPI-C Communication
- Platform specific development libraries for SNA & TCP/IP communication.

We will base the selection on an analysis of the data and requirements during the Blueprint phase of the project. The analysis will determine what data is in the legacy system and which SAP applications correspond to the business objects to be integrated.

Once we identify the technique, development team member and/or functional team member will prepare specification document.

4.6. Develop interface programs/objects and complete unit test

Once the interface program development is complete, developer will perform unit test based on the test scenarios that were specified in the specification document.

The organizational structure and actual customization affect the interface development; therefore, it is important to finalize or freeze customizations before the last test run. Customizing changes after the unit testing may result in additional fields that require the preparation and exchange of more data.

4.7. Complete Integration test

Perform integration test based on the test scenarios that were developed by functional team members for each interface object. Integration test scenarios might include background job scheduling, Back-up/recovery of data files and FTP process based on interface development requirements.

4.8. Data Validation

The Data Validation phase involves the verification of data to check accuracy, integrity, and reliability. Data validation will help support the quality assurance process for complete integration.

4.9. Perform Reconciliation Process

The reconciliation process involves verifying the reconciliation reports and executing programs to verify data. The reconciliation process includes the verification of total records between the systems, the validation of data elements, and the verification of financial information. The following reconciliation processes help validate that data conversion is successful:

- **Manual Validation Using Online System:** After the data is exchanged between two systems, functional team members for data accuracy will do a manual validation. As an example, simply checking a GL account or balances in hand for a few items can validate inventory value balances. This will further help validate the correctness of the data loaded.
- **Reconciliation Reports:** Reconciliation reports are the checks and balances reports between the legacy system and SAP. Reconciliation reports can identify the number of records exchanged from the legacy system, the number of records imported into the CSES database, and the number of data exceptions.
- **End-User Verification of Data:** The end user verifies the data as a part of the user acceptance test. End users compare actual results with expected results and document any discrepancies. The data conversion team verifies the test results and makes appropriate data modifications.

5. SAMPLE ANALYSIS DATA GATHERED:

Interface ID:	
Proposal ID:	
Interface description:	

Filename: C:\Documents and Settings\edward.brotsky\My Documents\PDF Deliverables\1.1 Project Prep\Final\Technology\1.1.18_TECH_ST_Interface_r1.2.doc

Initiative:	
Inbound/Outbound:	
Application Name:	
Application - Tech. Information:	
File Format:	
Frequency:	
Spec. Time:	
Batch/Real-time:	
Dependencies:	
Avg. No. Of Records/Run:	
FTP requirements:	
Backup & Recovery requirements:	
Short Description:	
Long Description:	
Functional Contact (Name, Phone and E-Mail):	
Tech. Contact (Name, Phone and E-Mail):	
Sample Data file (File Name and Location):	
Record Structure (File Name and Location):	
Business Process	
SAP T. Code	
Questions:	
Comments	

Related links:

SAP Home page

- <http://www.sap.com>
- SAP Interface Home page
- <http://ifr.sap.com>
- SAP Help portal – Online documentation containing details of all the components
- <http://help.sap.com>
- Comprehensive list of all BAPI, RFC & IDoc message types available in R/3
- <http://ifr.sap.com/catalog/query.asp>
- SAP Connectors (at the SAP Service Marketplace)
- <http://service.sap.com/connectors>
- General whitepapers and brochures from SAP AG
- <http://sap.com/solutions/technology/brochures.asp>

EAI Products for SAP R/3

- <http://www.sap.com/solutions/exchanges> : mySAP Exchanges
- <http://www.crossworlds.com> : IBM Websphere / Crossworlds
- <http://www.bea.com/products/weblogic/integration> : BEA Weblogic Integration
- <http://seebeyond.com/products/egateOverviewProducts.asp> : Seebeyond 'e*Gate'
- <http://www.it-architects.com/sapproducts.html> : IT Architects 'Connect-IT'
- <http://www.tibco.com/products/ax/index.html> : Tibco Business Connect
- <http://www.btalk.com> : Backsoft's 'btalk'
- <http://www.tsisoft.com> : Mercator Integration Broker

Other EAI - SAP related Journals and websites

- <http://www.EAIJournal.com>
- <http://eai.ittoolbox.com>
- <http://www.intelligentEAI.com>
- <http://eai.ebizq.net>
- <http://www.itpapers.com>

- <http://searchsap.techtarget.com>